1. Weighted Least Squares (textbook §11.1)

Recall regression model \( Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_{p-1} X_{p-1} + \epsilon \) in matrix form:
(Ch. 5, Handout #6.b p. 2)

\[
\begin{bmatrix}
Y \\
\end{bmatrix} = \begin{bmatrix}
X_1 \\
\vdots \\
X_p \\
\end{bmatrix} \begin{bmatrix}
\beta_0 \\
\beta_1 \\
\vdots \\
\beta_p \\
\end{bmatrix} + \begin{bmatrix}
\epsilon \\
\end{bmatrix}
\]

Model assumption: \( \epsilon \sim N(0, \sigma^2 I) \)

- If constant variance, (i.e., \( Cov(\epsilon) = \sigma^2 I \)), then use OLS:
  \[
  b = (X'X)^{-1}X'Y
  \]

- If non-constant variance, then can estimate and account for it (WLS):
  \[
  V = Cov(\epsilon) \quad \text{(typically assumed diagonal)}
  \]
  \[
  W = V^{-1}
  \]
  \[
  b_w = (X'WX)^{-1}X'WY
  \]

Typically, \( Cov(\epsilon) \) must be estimated

- can often relate variance of residuals (or squared residuals) to predictors or \( \hat{Y} \) values

- example (as in Ex. 1 of Handout #11.a): residual vs. \( X_1 \) is megaphone-shaped
  (linear relationship between SD of residual and \( X_1 \))
  - regress absolute residuals on \( X_1 \) and get predicted values \( s \) (as function of \( X_1 \))
  - define weights \( w = 1/s^2 \)

- see p. 425 for other examples
  - key is how to estimate \( w \) for given scenario, as a function of \( X \)’s
2. Ridge Regression (textbook §11.2)

Using OLS matrix form of linear regression (from Ch. 5, as above):

\[ b \sim N(\beta, (X'X)^{-1}\sigma^2) \]

- Each coefficient estimate \( b_k \) has an expected value of \( \beta_k \); it’s unbiased
- (Current data are one sample; consider repeating sample collection many times, each time fitting the model and obtaining an estimate \( b_k \) for each \( k \); the distribution of these is the sampling distribution for the estimator \( b_k \), with mean \( \beta_k \) and variance the corresponding diagonal element of \( (X'X)^{-1}\sigma^2 \); normality assumed)

Recall that understanding & diagnosing multicollinearity is facilitated by “standardizing” variables (recall pp. 3-4 of Handout #7.b; here, “correlation transformation”):

\[
\begin{align*}
Y^* &= X^*\beta^* + \varepsilon \\
b^* &= (X'^*X^*)^{-1}X'^*Y^* \\
Cov(b^*) &= (r_{XX})^{-1}\sigma^2 \\
Y^*_i &= \frac{1}{\sqrt{n-1}} \left( Y_i - \bar{Y} \right) \\
X^*_k,i &= \frac{1}{\sqrt{n-1}} \left( X_{k,i} - \bar{X}_k \right) \\
r_{XX} &= \text{correlation matrix of } X's \\
r_{YY} &= \text{correlation vector between } Y \text{ and } X's
\end{align*}
\]

Recall symptom of multicollinearity: inflated variance of \( b_k \) estimates (diag. element of \( Cov(b^*) \))

- Can we reduce the variance of estimates by biasing them?
- Think of two sampling distributions: (sketch)
  1. unbiased estimate (with inflated variance)
  2. biased estimate (with reduced variance)

Allowing estimates (the \( b_k \)) to be slightly biased could reduce their standard errors \( \rightarrow \) avoid major problem of multicollinearity. How to introduce bias and reduce variance?

- introduce small, positive biasing constant \( c \geq 0 \):
  \[ b^R = (r_{XX} + c \cdot I)^{-1}r_{YY} \]
- We call \( c \) the “ridge parameter”
  - \( c = 0 \) gives unbiased estimates: OLS
  - larger \( c \) gives larger \( b \), smaller \( \text{variance} \)
Two graphical summaries to choose the “right” ridge parameter $c$:
(Note: these are guides; there is no “optimal” decision)

1. Ridge Trace Plot
   - (Need standardized data for this to be meaningful; SAS does internally)
   - Simultaneous plot of $b^R_1, \ldots, b^R_{p-1}$ (using standardized data) for different ridge parameters $c$ (usually from 0 to 1 or 2)
     
     - As $c$ increases from 0, the $b^R_k$ may fluctuate wildly and even change signs
     - Eventually the $b^R_k$ will move slowly toward 0

2. VIF Plot
   - Simultaneous plot of the variance inflation factor for the $p - 1$ predictors for different ridge parameters
   - As $c$ increases from 0, the VIF drop toward 0

In general, choose smallest ridge parameter $c$:

1. where the $b^R_k$ first become “stable” (their approach towards 0 has slowed)
2. and the VIF’s have become “small enough” (close to 1 or less than 1)

Why is it called “ridge” regression?

- $c$ is added to diagonal of $r_{XX}$ → makes sort of ridge there
- ridge trace plot shows how each $b^R_k$ follows a ridge for increasing $c$

Ridge regression is a type of “shrinkage” method:

1. Impose a “penalty” $\lambda$ on the sizes of the $b_k$’s
2. Find $b$ to minimize

$$\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 + \lambda \sum_{k=0}^{p-1} (b_k)^2$$

(First part is OLS; second part is penalty)

3. Which penalty ($\lambda$)?
   - same approach as with selecting ridge parameter $c$
4. The penalty ($\lambda$ or $c$) effectively “shrinks” the $b_k$ towards 0
5. Another popular shrinkage method: the Lasso
   - minimize $\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$ subject to $\sum_{k=1}^{p-1} |\beta_k| \leq \lambda$
   - side note: “adaptive” LASSO allows weight on each $\beta_k$ in optimization criterion
A few extra notes on ridge regression:

- choice of ridge parameter is somewhat subjective, but must be defendable
- given ridge parameter \( c \), can get resulting parameter estimates \( \hat{b} \) on the “unstandardized” (original data) scale
  
  - SAS gives these automatically, but need textbook equation 7.46b to get intercept \( \hat{b}_0 \):
    
    \[
    \beta_0 = \bar{Y} - \beta_1 \bar{X}_1 - \beta_2 \bar{X}_2 - \ldots - \beta_{p-1} \bar{X}_{p-1}
    \]

- ridge regression estimates \( \hat{b} \) tend to be more robust against small changes to data than are OLS estimates
- ridge regression tends to produce more precise predicted values (\( \hat{Y} \)) than OLS when predictors are highly related; also better at slight extrapolation
- predictors with very unstable ridge trace (tends toward zero without any plateau or slowing down) may be dropped from model; also consider dropping predictors with very small ridge trace

- major limitation: traditional inference is not directly applicable to ridge regression estimates (be cautious)
  
  - need “bootstrapping” to evaluate precision of ridge regression estimates
  
  - (computationally intensive, beyond scope of this course)

- Why do ridge regression? Maybe want to adjust for multicollinearity but still keep certain predictors in model (for mechanistic theory) with “correct” sign coefficients (example: thigh)

  Ex: thigh is pos. corr. w/ body
  but OLS coeff was neg.
  . Ridge: reg coeff on thigh was pos.
3. Robust Regression (textbook §11.3)

When model assumptions met, OLS is “best”; if not, consider alternatives. Here, consider dealing with influential observations or outliers.

IRLS (iteratively reweighted least squares)

1. Obtain (maybe from OLS) \( b \), then calculate \( \hat{Y} = Xb \) and \( e = Y - \hat{Y} \)
2. Calculate weights \( W \), based on \( e \) (lots of weight functions available)
3. Calculate (WLS) \( b_w = (X'WX)^{-1}X'WY \) and resulting \( e = Y - Xb_w \)
4. Iterate steps 2 & 3 to convergence of \( b_w \)

How to calculate weights?
- usually chosen to optimize some criterion
- the choice of criterion determines the method of weight calculation

M-estimation

- If \( u_1, \ldots, u_n \) are iid from some distribution with parameter \( \theta \), then the type-M estimate of \( \theta \) is of the form
  \[
  \hat{\theta} = \operatorname{arg\ min} \sum \rho(u_i; \theta) 
  \]
  where \( \rho \) is some “scalar objective function”

- Example: \( \rho(u; \theta) = -\frac{1}{n} \log f(u; \theta) \), \( f \) is pdf of distribution of \( u_1, \ldots, u_n \). Then
  \[
  \hat{\theta} = \operatorname{arg\ max} \sum \frac{1}{n} \log f(u_i; \theta) 
  \]
  \[
  = \operatorname{arg\ max} \text{ (likelihood)} 
  \]
  \[
  = \text{(what is this called?)} 
  \]

- W-estimation approach in IRLS:
  1. Calculate robust estimate of \( \sigma \), such as \( s = \frac{\text{MAD}(e)}{0.6745} \)
  2. Let \( u_i = \frac{e_i}{s} \) be “scaled” (or standardized) residual
  3. Calculate (diagonal) weights \( w_i = \frac{\psi(u_i)}{u_i} \)
    - where \( \psi(u) = \rho'(u) \) for some scalar objective function \( \rho \)

Example – Tukey Bisquare (sometimes called Tukey’s Biweight):
\[
\rho(u) = \begin{cases} 
\frac{c^2}{3} \left(1 - \left[1 - \left(\frac{u}{c}\right)^2\right]^3\right) & \text{if } |u| \leq c; \text{ default } c = 4.685 \\
\frac{c^2}{3} & \text{otherwise}
\end{cases}
\]

Bisquare weight function: \( w(u) = \left(1 - \left(\frac{u}{c}\right)^2\right)^2 \) for \( |u| \leq c \), 0 otherwise

Note: M-estimation works well for outliers; for leverage points, use MM-estimation (see SAS help)

What if $Y$ vs $X_1, \ldots, X_{p-1}$ not linear (in $\beta$’s)?
– Usually need mechanistic theory

Example 4.1: $Y = \beta_0 + \beta_1 X_1^{\beta_2} - \beta_3 \exp(\beta_4 X_2) \ (+\epsilon)$ (with simulated data)

**proc nlin** fits these nonlinear models

- Parameters estimated by an iterative process to reduce the SSE at each iteration, until convergence
- Keys to [useful] convergence:
  - form of nonlinear equation
  - initial parameter estimates

Example 4.2: a nonlinear curve to describe sand compression, from Lagioia et al. (1996)
Computers and Geotechnics 19(3):171-191

$$f = \frac{p}{p_c} - \left( \frac{q}{p_M \cdot k_2} \right)^{\frac{k_2}{(1-p)(k_1-k_2)}} \left( \frac{\eta}{\left(1+\frac{q}{p_M k_1}\right)^\frac{k_1}{(1-p)(k_1-k_2)}} \right),$$

where
- $f$ = yield surface (response)
- $q$ = deviatoric stress (predictor)
- $p$ = mean effective stress (predictor)
- $p_c$ = hardening / softening constant defining current size of surface (known)
- $\eta$ = stress ratio $p/q$
- $M$ = parameter defining value of $\eta$ with no strain increment
- $\mu$ = parameter defining general slope of $d$ vs. $\eta$ curve
- $\alpha$ = parameter defining how close to $\eta = 0$ axis curve bends towards $d = \infty$
- $d$ = dilatancy, $2\mu M(1-\alpha)$

Goal: find $\mu$, $\alpha$, and $M$ to make $f \approx 0$, and look at the relationship between these three parameters

**proc model** estimates such nonlinear systems (can do multiple equations)

From playing with this in SAS, it appears that to achieve convergence of estimates in **proc model**, the most important thing is that at least one of the tails of the $q \times p$ curve to be fit has data along most of it. To make the convergent estimates “good”, it appears necessary to have data along both tails. It is also crucial that the initial starting estimates be good, especially for $M$ (maybe within .2 or so).